Innovating education with an educational modelling language

Citation for published version (APA):

Sloep, P., Van Bruggen, J., Tattersall, C., Vogten, H., Koper, R., Brouns, F., & Van Rosmalen, P. (2006). Innovating education with an educational modelling language: two case studies. *Innovations in Education and Teaching International*, *43*(3), 291-301. https://doi.org/10.1080/14703290600750838

DOI:
10.1080/14703290600750838

Document status and date:
Published: 21/11/2006

Document Version:
Peer reviewed version

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:

https://www.ou.nl/taverne-agreement

Take down policy
If you believe that this document breaches copyright please contact us at:

pure-support@ou.nl

providing details and we will investigate your claim.

Downloaded from https://research.ou.nl/ on date: 05 May. 2023

Open Universiteit
www.ou.nl

TITLE

Innovating education with an educational modelling language: two case-studies

Peter B. Sloep, Jan van Bruggen, Colin Tattersall, Hubert Vogten, Rob Koper, Francis Brouns, Peter van Rosmalen

Open Universiteit Nederland

All correspondence should be sent to the first author:
Dr. P.B. Sloep
Open Universiteit Nederland
Postbus 2960
6401 DL Heerlen
The Netherlands
<mailto:peter.sloep@ou.nl>

SUMMARY

The ultimate aim of this study is to investigate how one may maximise the success chances of an educational innovations based on the implementation of an educational modelling language. These languages are both technically and organisationally quite demanding. Their implementation therefore constitutes a major innovation. Two different implementation cases of the educational modelling language EML provide the data for the investigation. The one case is situated in an institution for higher professional education that caters for on-campus students, the other in an institution for higher, open distance education that serves off-campus, home-based students. Together, the cases represent two important dimensions of the space of possible educational modelling language implementations. Rogers' diffusion-innovation theory is used as the backdrop for the analysis of the cases. It not only provides a common yard stick, but more importantly helps understand why the implementations failed in particular respects. Thus, it helps formulate guidelines for future implementations. In this respect, a current candidate, IMS Learning Design, is discussed at some length.

Keywords

innovation; innovation-diffusion theory; Rogers; education; virtual learning environment; VLE; educational modelling language; EML; learning design; IMS Learning Design specification; IMS LD; evaluation

1 INTRODUCTION

Long since, formal education has been almost synonymous with classroom teaching. The predominant pedagogic approach amounted to the teacher reciting what students were supposed to learn. Over the last decade this situation has changed significantly. With the advent of the computer, the classroom learning environment has been extended with, in some cases even replaced by a virtual learning environment (Sloep, 2004a; Oppenheimer, 2003). This learning environment is virtual in that the learning affordances offered by the traditional classroom have been replaced by and augmented with computer mediated affordances.

Communication among students, for instance, which in a classroom takes place in virtue of the students' physical proximity, is now mediated via e-mail, chat programs, conferencing software, and the like in the computer realm. Similarly, learning content that in a classroom would either be orally transmitted by the teacher or read by the students in books, lecture notes, etc., in the computer age is made available also or solely on screen. Moreover, as an extension of services offered to the student, teacher's help may be accessed even outside office hours, be it via e-mail or via compiled lists of frequently asked questions (Salmon, 2000).

Over the last 5 years or so, the virtuality of these kinds of learning environments has increasingly become supported by dedicated software tools, aptly called Virtual Learning Environments (VLEs). The VLEs of the current generation typically mix the structure of the pedagogical message (the pedagogical model or educational scenario) with the content of the message. They are also very much modelled on current teaching practices and thus tend to preserve these rather than stimulate the emergence of novel ones (Emans et al., 2004) Although there may be not much wrong with current practices, redoing with an ICT supported environment what we already did without, can hardly count as a justification of the huge sums of money we spend on ICT in education. Minimally, increases in efficiency, effectiveness, and attractiveness are needed. It is because of these kinds of reasons that the Open University of the Netherlands started to work on an Educational Modelling Language (EML) (Koper, 2001). Briefly, with such languages one diverges the educational content from the pedagogy. The language allows one to describe the 'learning flow' in a notational system (a modelling language, if you like). The language really is a meta-language in that it is capable to describe a large variety of different scenarios (Koper & Olivier, 2004; Tattersall et al., 20005; Vogten et al., 2005; Wilson, 2005). Although much of course depends on how one uses such a language (with natural language both good and bad poetry can be written), there is at least the potential to improve efficiency, effectiveness, and attractiveness. For more on EML and its successor IMS Learning Design (LD), please consult Koper et al. (2003) and Koper and Tattersall (2005).

The ability actually to work with an educational modelling language very much depends on the availability of scenario editors, content editors, validators to check syntactic and semantic conformance, repositories for storing the scenarios and the content, players that dynamically generate the html that a browser interprets, not to mention (application) interfaces to portals, administrative systems of various kinds, authentication and authorisation modules, etc. (Koper & Tattersall, 2005). The use of these tools, in turn, requires a suitable organisational infrastructure (Schlusmans, Koper & Giesbertz, 2004). Although ideally the tooling follows the organisational demands, in the early days of an innovation's dissemination one has to make do with the tools that are available, however user-unfriendly and however alien to existing organisational practices they may be.

Although the EML specification was published at the end of 2000 only, many experiments, including pilot implementations, had already been conducted. Work on the IMS Learning Design specification begun in 2001, it was published in February 2003 (IMS, 2003). This has sparked a flurry of activities, such as the foundation of the Valkenburg group as a means of developing LD related tooling, the funding of the UNFOLD LD dissemination project, the start of a number of PhD projects, the writing of a book on the practicalities of LD usage (Koper and Tattersall, 2005), the creation of open source LD editors (the Reload project) and runtime players (Coppercore). (see <www.learningnetworks.org> for details). All these activities will ultimately result in LD implementation projects.

This raises the question of how the chances successfully to implement these kinds of innovation projects can be maximised. We are dealing here with major projects, that draw upon vast financial and organisational resources. The stakes, therefore, are high: failure is hardly an option. It is thus a matter of much urgency to formulate guidelines that help these projects to meet with success. We will look into two cases here that both were attempts to implement EML. From their analysis, we shall attempt to extract some useful guidelines. We will use Rogers' innovation-diffusion theory to inform our analysis (Rogers, 2003). Section 1 discusses the theory's most relevant tenets. After a brief summary of the methods used in two case studies

(Section 2), we provide an overview of the studies' findings that are relevant in the context of our Rogersian analytical framework (Section 3). We conclude the paper by formulating and discussing a number of guidelines (Section 4).

2 ROGERS' INNOVATION DIFFUSION THEORY

To Rogers (2003), diffusion of an innovation is a communication process, which involves both senders (the change agents) and receivers (a community). Here, we are particularly interested in the receiving community. Communities, according to Rogers, have a characteristic structure, with *norms* that may make acceptance more or less likely, with *opinion leaders* who may advocate the innovation or rather do the opposite, with a homogeneous or heterogeneous *opinion structure*, etc. The opinions of individual community members are influenced by their fellow community members. Some will adopt almost immediately (early adopters), others will follow in their footsteps quickly (early majority), or late (late majority), yet others (laggards) will resist until the last moment. However, ultimately every community member makes up his or her own mind. According to Rogers, their decision to adopt or reject is made on the basis of five *innovation attributes*. Crucially, as an innovative product always supersedes something in use, they will use the product in use as their point of reference.

• *Relative advantage* What are the benefits of the innovation in economics terms, in terms of social prestige? With respect to innovations that are top down imposed by the management, are there any incentives for adopting? The bigger the relative advantage is perceived to be, the better the chances of the innovation's adoption.
• *Compatibility* How does the innovation fit in with existing community and personal values, with past experiences and needs? Innovations that are disruptive in some sense stand less of a change to be widely adopted.
• *Simplicity* To what extent is an innovation perceived as simple or difficult to understand and use? The simpler an innovation, the better its adoption chances.
• *Trialability* To what degree can the user experiment with the innovation? Trialability helps adoption as it confers a degree of control on the user. Trialability particularly

helps swing groups that adopt early, it is immaterial to the laggards, who mostly tend to simply follow their predecessors.

• *Observability* How visible are the beneficial results of an innovation? Particularly innovations that have obvious, clearly visible benefits, are more easily adopted.

3 MATERIALS AND METHODS

The two case-studies discussed here were the first experiences with full-blown implementations of EML-Edubox outside the group of its original developers. They were set up to collect recommendations for improving both the EML specification and the Edubox player. (Edubox is a software application that has been developed to manipulate valid EML(XML)-code in such a way that a user – learner, tutor - is able to interact with it via a graphical user interface.) So the evaluation reports paid most attention to distinguishing between problems and issues that could be attributed to EML or to Edubox, rather than to failing organisational measures. The results of these evaluations have been collected in internal reports. Our interest in this paper is different, though. Here, we are primarily interested in the question of how the user communities reacted to the introduction of EML-Edubox. Thus we've picked and chosen from these reports those results that are relevant to our present purposes, rather than report them integrally.

Both case studies share a number characteristics. We'll discuss these first, before turning to the differences. In either case, courses were created by coding XML instance documents. These documents contain both the educational content and the learning design. Instance documents were developed by teachers, instructional designers, and EML experts jointly. Almost always, the instructional designer and the EML expert were the same person. In case 1, the EML designers were quite experienced and hired by the school. In case 2, most of the EML designers were employed by the school and had received prior EML training. They were not experienced, but had access to experienced EML designers. They also frequently convened to discuss issues they ran into.

When the development phase was completed, the courses were published on the Edubox system. It ran on a remote server that had to be accessed through the Internet via a browser. After publication, students and teachers could access the course in a similar way.

Case 1 involves 3rd year students at an institute for higher professional education who took a course on how to develop a business plan. The course spanned a period of 10 weeks, the first five of which were mainly devoted to lecturing and making assignments individually, the second five to collaboratively, in groups of eight students producing the business plans. The materials the students needed for the course were both available in paper and digital formats.

EML-Edubox was used to support the collaborative work of the students, although it had also been available during the first five weeks. Per group of 8 students, 2 computers were available. The computers for all groups were situated in one large classroom. At the staff side, originally 4, later on only 2 developers were involved in the course; during delivery, 8 tutors were available. Each tutor had a specific task. Both developers were subjected to a semi-structured interview. Tutors and students received the same questionnaire (35 questions, 31 closed, 4 open), which was distributed via e-mail. Only 3 out of 8 tutors actually filled out the questionnaire. The 5 who did not, had not made use Edubox at all. 31 out of 58 students filled out their questionnaires. The Edubox log covers 52 students and 3 teachers. Only the questions that are relevant to the reception of EML-Edubox by the community will be reported here.

Case 2 concerns a university for distance education. The focus of this case is on the course development process. Therefore, only opinions of staff, not of students were gauged. Although a variety of quantitative data were collected, none are directly relevant in the present context. Therefore, only qualitative results will be reported, garnered from the report. The project of case 2 aimed to ‘transform’ the contents and educational scenario’s of 6 courses to befit the EML-Edubox learning environment. This meant extracting the content from an existing system and explicitly describing the learning flow. This course transformation was carried out by staff teams. Each team consisted of a content expert (teacher), and instructional designer and EML expert. Teams convened regularly to discuss transformation problems and issues; they then had access to an experienced EML expert. The school management had furnished the project teams with the time and means to carry out the transformation projects.

TABLE 1 about here

4 RESULTS

4.1 RESULTS FOR CASE 1

The structured interviews with the tutors led to the following:

1.1 The teachers were not properly trained for the job, nor were they properly facilitated.

1.2 Originally, a team of 4 teachers was supposed to carry out the development work. Due to a variety of circumstances, the team was rapidly culled to consist of 2 persons only. They felt marginalised.

Students and staff filled out a questionnaire. They were allowed to add comments, two of which are relevant in this context (numbers again refer to page numbers):

1.3 The benefits a virtual learning environment may have to offer could not be reaped as EML-Edubox was mainly used for electronic page turning. Besides, all the texts were also available in textual form, further decreasing Edubox's potential usefulness.

1.4 Frequent face-two-face contacts among the students and between students and tutors - for example for carrying out group work - reduced the need to use a virtual learning environment such as EML-Edubox.

1.5 The frequent face-to-face contacts also meant that in every group only one person really needed to consult Edubox. That person became the expert, almost to the point of eliminating the need for others to work with Edubox.

The remaining results are all derived from the questionnaire and are reported in tabular form (Table 2).

TABLE 2 about here

4.2 RESULTS FOR CASE 2

The following results that are relevant for our present purposes were:

2.1 No user manuals, examples, and demos were available; the technical EML Reference Manual came available only late in the project.

2.2 Development teams could not use their own computers but had to resort to computers at a central location.

2.3 The team members had no previous experience in working with EML-Edubox and hence required training, which was a little late but otherwise adequate.

2.4 Support from the faculty management differed considerably between projects.

2.5 EML editor was user-unfriendly.

2.6 EML data-entry work is boring and likely to cause RSI in the long run.

2.7 EML-Edubox is not market-ready.

2.8 EML-Edubox does not add much to what is already available; indeed in many respects it means a step backwards.

2.9 EML may well not become an accepted standard, which puts our school in a difficult situation.

2.10 It should be possible quickly to preview the results of the EML coding.

2.11 In 5 out of 6 cases, little more than a straight-forward transfer - in terms of educational design and contents - was carried out. (Case 6 was not completed at the time of the evaluation.)

4.3 APPLYING THE ROGERSION FRAMEWORK

The members of the communities of cases 1 and 2 may be classified as early adopters and early majority, respectively (Rogers, 2003). In case 1, there was no previous experience with EML-Edubox whatsoever, in case 2 pilots had already been conducted, with other individuals. Therefore, while in school 1 staff was largely ignorant with respect to EML-Edubox, in school 2 staff was likely to have some opinion on its adoption. However, no outspoken opinion leaders participated, so nobody voiced either a strong commitment or strong objections to the deployment of EML-Edubox. In both schools deployment was very much a management decision. Although the participating staff people could volunteer to participate, some at least did not do so wholeheartedly. In neither case, staff were given incentives to participate. The participants' opinions therefore seem to be largely their own, formed on the basis of the innovation attributes Rogers mentions. It is in this respect relevant to note that school 1 had little to no previous experience with the use of VLEs. EML-Edubox offered both students and staff their first experience with a VLE. In school 2, on the other hand, a VLE has been in use for about 5 years. This system had been built locally and had been steadily expanded. It had been intended as an extension of the largely paper based course distribution system in use. Over the years, however, elements of paper based courses (notably workbooks) had been migrated to the virtual learning environment. They were implemented there as straightforward html-pages. Gradually, a common page lay-out and menu structure had arisen. Thus a system had emerged that offered functionalities similar to what commercially available VLEs offered. The inability to do version control, the awkwardness of adapting courses, and the labour intensity of developing Java-script based customised functionalities were all arguments that led the school’s management to the decision to deploy EML-Edubox.

Relative advantage

The most important benefit of the use of EML-Edubox is that one may use a variety of pedagogic models with the same VLE, also models that are hard to implement in currently available VLEs. The models and resulting designs developed in school 2 are all relatively simple (result 2.11). And, not surprisingly, people at school 2 complained that EML-Edubox added little value (2.8). School 1 did adopt a non-linear model, with the necessity for collaborative work. However, the way it was implemented meant that the deployment of EML-Edubox added little value (results 1.3 - 1.5). Also, the management in school 1 did little to provide incentives for those participating in the project (results 1.1 & 1.2). So in both cases the perceived relative advantage was low.

Compatibility

For school 1 the compatibility issue did not arise in the context of their use of Edubox (1.6-1.10), but was restricted to the way the EML development work was organised (see also the methods section). School 1 is a rather traditional school in the sense that content development is carried out by teachers themselves, who rely on their own experiences and, to a large extent, on commercially available text books. For lack of user-friendly EML-editors and local EML expertise, the development process had to be organised by bringing in EML experts from outside. Also, as the layered instantiation process - from pedagogic model via course to running course - benefits from the participation of instructional designers, these were also brought in. All of this was alien to what was customary at school 1.

The case of school 2 is different. There, the development process had always been characterised by a division of labour between content experts, instructional designers, and graphical designers. The addition of XML experts to the development team thus was compatible to what one was used to. Nevertheless, many complaints were voiced, varying from the lack of manuals, training, user-friendly editors and management support (results 2.1, 2.3, 2.4 - 2.6) to doubts about the external viability of the EML-Edubox system (results 2.7 and 2.9). Superficially, these merely point to technical defects, which does constitute a violation of the user's expectations. However, they also point to a more deeply seated, less easily resolvable problem. As explained, the VLE in use at school 2 made use of html-pages that in some cases were enriched with Java scripts. These pages were not stored in and served by a content management system; therefore, they were simply accessible as html by the teachers. Thus they could easily make adjustments to a running course, on the fly, that is. The way Edubox operates may be compared to a content management system in that access rights may be set and limitations may be imposed on the degree to which a running course can be altered. So teachers can only with difficulty alter courses in exploitation. In this respect, the introduction of EML-Edubox in school 2 amounted to a clash with hard-won existing practices.

Simplicity

The use of Edubox is not particularly complex, at least not more complex than the use of any other VLE. This is reflected in the responses obtained (results 1.11 – 1.16). The development process, however, is quite complex (results 2.1, 2.3 2.5). As described above, it involves teams of specialist and the use of specialist software, like generic XML-editors. Presumably, the availability of graphical EML-editors would change all this, much as the availability of dedicated html-editors has allowed html-editing for the masses. However, at both schools, one had to work with powerful, yet user-unfriendly generic XML-editors. All this amounted to a relatively complex rather than simple innovation.

Trialability

The Edubox application may be test-driven easily. For that purpose, in both cases a dummy course was made available. That way, users could experiment with the application's interface and feature set. However, one significant obstacle remained. Content developers were eager to repeatedly test-drive courses at the various stages of their development (compare with viewing a web page developed with a textual html-editor in a browser). So much became apparent in discussions with them (result 2.10). Although this is possible in principle, it is a time-consuming and cumbersome process, involving the help of specialists (result 2.2). Perhaps the need for trying out a course during its development wanes with increasing experience in the content development process, but for the developers in both schools this lack of trialability was a serious obstacle.

Observability

In case 1 the lack of any necessity to use EML-Edubox extensively did obviously not contribute to its visibility. Benefits its use may have brought to students as well as teachers could not become apparent, simply because only few students (results 1.4 and 1.5) and few teachers (result 1.2) were in fact using it.

There is a further consideration. Re-use of pedagogic models and learning designs as well as version control of published courses are features that are beneficial to an organisation as a whole. Something similar goes for medium neutrality, a feature which is inherent to all XML files and thus to EML. It means that the medium to which an XML file is going to be published may largely be decided on after its development. Clearly this is beneficial to an organisation, that may decide to publish the same material both in say printed and electronic form. Such benefits, however, fail to entice individual teachers, let alone students. They only become apparent to them in indirect ways and after a while: the money or time an organisation saves may subsequently be spent to their benefit; or, the ability to deliver courses both in printed and in electronic form constitutes an increased level of service to the students. The observability of the innovation is thus low.

5 CONCLUSION AND DISCUSSION

The results make it abundantly clear that the introduction of EML-Edubox i) had a low perceived relative advantage over existing practices, ii) led to a significant change in existing practices (low compatibility), iii) amounted to a complex rather than simple innovation; in addition to this, it was difficult to test-drive designs and the benefits of the use of EML-Edubox accrue to the organisation at large rather than the individuals who bear the brunt of the innovation; finally, both trialability (iv) and observability (v) were low. In summary, then, the introduction of EML-Edubox had little change of being successful, and this matches the tone and conclusions of the evaluation reports. However, could one have avoided this conclusion, that is, is a successful introduction of this kind of innovation at all possible?

The low relative advantage results at least in part from the novelty of the EML-Edubox system. No doubt, once teachers and educational developers have become more experienced, they will develop pedagogical models that are more exciting and hence are a testimony to the system's power. But for them to become more adroit at using EML-Edubox, they first need to adopt it. To kick-start them, one may provide templates (Sloep, Hummel & Manderveld, 2005) of innovative models, allowing them quickly to develop exciting and thus enticing educational models.

The compatibility issue is a little harder to tackle. School 1, it seems, used the introduction of EML-Edubox as an occasion also to introduce a division of labour were there had been none before. In school 2 the introduction of EML-Edubox meant a re-establishment of old customs (the inability to edit on the fly) that many probably felt glad to have gotten rid off. In both cases, then, organisational habits had to change, something which is bound to spawn resistance (Noble, 1998 bears witness to this). The need for a division of labour as well as the inability to edit on the fly may well be remedied by technical means in due time. It appears, however, that both changes were made by the respective managements for reasons that go beyond the wish to introduce EML-Edubox, to wit, to introduce a division of labour in school 1, and to install a less haphazard change management policy in school 2. If so, there is little the introduction of EML-Edubox *per se* can be blamed for other than that EML-Edubox allows these kinds of organisational changes to be made. To increase the success changes of a technological innovation, implementations should therefore avoid stacking unnecessary organisational changes on top of technological ones.

The advent of user-friendly, dedicated EML editors (cf. Brouns, 2003), preferably ones that can be fed with templates of didactic models (cf. Sloep, Hummel & Manderveld, 2005) will help make the development of EML-based courses much simpler. If the contribution that user-friendly html-editors have made to the success of 'the web' (Berners-Lee, 2000) may be taken as a yard stick, the simplicity issue is indeed not one to worry about very much. Similarly, technical advances should allow users (developers, teachers) simply to test-drive a course while still under development. Then trialability should not been an issue anymore either.

As with compatibility, the ability to observe the benefits of the introduction of EML-Edubox is not a technical but a management issue. If management succeeds in showing how the introduction of EML-Edubox allows everybody - staff and students alike - to enjoy better educational quality, then the likelihood of adoption is increased significantly.

In retrospect, these conclusions should not have come as a surprise had we taken into account the Rogersian notion of a stage. Rogers (2003) distinguishes five phases in an innovation's adoption and identifies them by the people involved: innovators, early adopters, early majority, late majority and laggard. Nolan (1973) distinguishes four stages in the use of (mainframe) computers in organisations: initiation (a few enthusiast start up something new), contagion (colleagues become 'infected' and join in), control (management tries to contain and streamline the novelty), and integration (the novelty has become main stream). The Gartner Hype Cycle alludes to a similar kind of process (Gartner, 1995). The point to take is that innovations at some juncture are not the prerogative of a few enthusiasts any more but become institutionalised. This demarcation point lies between Roger's early adopters and early majority, or between Nolan's contagion and control. In managed innovation projects, this demarcation point marks the transition from development to implementation, from a management that fosters diversity to a management that culls it.

Perceptions of a project's relative advantage, compatibility, complexity, trialability and observability take its intended use as a benchmark. When institution-wide deployment is the goal, the deployment community rightly make high demands. If the innovation does not seem do them any good, is ill-compatible, complex and not trialable, they will only adopt if they are properly compensated. If, on the other hand, the immediate goal is more limited, if it is clearly communicated to the intended users that the goal only is improving an existing prototype rather than large-scale adoption, then they are likely to adopt a less demanding benchmark. They will be willing to put up with a low relative advantage, low compatibility, low trialability, and complexity.

The EML-Edubox project as reported here clearly was still in its development phase. Indeed, its developers approached schools 1 and 2 to help them find and remedy problems and issues with the current EML-Edubox implementation (see section 1).

Many of the problems the project ran into resulted from it not being properly treated as a development project, from bringing in wider organisational concerns that really belong to the realm of the implementation stage. The messages sent out to the intended users vacillated between 'this is a development project, please provide us with feed-back' and ' this is a first implementation, you had better get used to the new workflow'. In view of this, it is no wonder that the project wasn't an unequivocal success: it was unclear to the user community what benchmarks to espouse.

Can these conclusions be generalised to cover implementations of the IMS Learning Design specification? Although IMS LD is a direct descendant of EML, there are some differences. LD does not contain a content model, users are advised to use XHTML; it does not contain provisions for structuring assessments, users are advised to use the IMS Question and Test Interoperability specification that may be absorbed ('name-spaced') into the LD specification'; it contains no metadata specification, users may adopt their favourite flavour of the IEEE LOM or the Dublin Core specification. Also, there is currently no equivalent of the Edubox system available yet, although the open source community is working both on LD capable editors (e.g. Reload, cf. < http://www.reload.ac.uk/> and players (e.g. Coppercore, cf. <http://coppercore.org/>). These differences, however, seem immaterial with respect to the above analysis.

Thus, when implementing LD and accompanying software (editor, content management system, or player), one should be clear about one's purpose. If LD-based software gets tested, inform the testing community clearly about the purpose and avoid at all cost to stack institution-wide changes on top of the test. If, on the other hand, the goal is to implement LD-based software on an institutional basis, heed Rogers' rules to the full.

REFERENCES

Berners-Lee, T. (2000) *Weaving the Web; The Past, Present and Future of the World Wide Web* London New York, Texere.

Brouns, F. (2003*) Editing EML 1.1 Preliminary survey of existing tools* Internal Report. Online available: <http://hdl.handle.net/1820/203>

Emans, B., A. Hondius, W. Koopal, S. Muizelaar, M. van der Wee, O. Wortman (2004) *ELO's in relatie tot onderwijsconcepten* [VLEs and their relation to educational conceptions]. Utrecht: Digitale Universiteit.

Gartner (1995) Understanding the Hype Cycle. Online available: <http://www4.gartner.com/pages/story/php.id.8795.s.8.jsp>

Hummel, H. G. K., Manderveld, J. M., Tattersall, C., & Koper, E. J. R. (2004) Educational Modelling Language: new challenges for instructional re-usability and personalized learning, *International Journal of Learning Technology* 1, 110-111.

Koper, R., B. Olivier, T. Anderson (2003*) IMS Learning Design Best Practice and Implementation Guide, Information Model, Information Binding.* IMS Global Learning Consortium Online available: <http://www.imsglobal.org/learningdesign/index.html>

Koper, R. (2001) Modeliing Units of Study from a Pedagogical Perspective; the Pedagogical Meta-model behind EML. Unpublished MS. Online available: <http://hdl.handle.net/1820/36>.

Koper, R., B. Olivier (2004) Representing the Learning Design of Units of Learning. *Educational Technology & Society* 7(3), 97-111.

Koper, R. & C. Tattersall (Eds) *Learning Design; A Handbook on Modelling and Delivering Networked Education and Training*. Heidelberg: Springer Verlag.

Noble, D.F. (1998) Digital Diploma Mills: The Automation of Higher Education, *FirstMonday* 3 (1). Online available: <http://www.firstmonday.dk/issues/issue3_1/noble/>.

Nolan, R.L. (1973) Managing the Computer Resource: A Stage Hypothesis. *Communications of the ACM* 16(7): 399 - 405.

Oppenheimer, T. (2003) *The Flickering Mind: The False Promise of Technology in the Classroom and How Learning Can Be Saved* (New York etc., Random House).

Rogers, E. M. (2003) *Diffusion of Innovations, fifth edition* (New York etc., Free Press).

Salmon, G. (2000) E-*Moderating - The Key to Teaching and Learning Online* (London, Kogan Page).

Schlusmans, K.H.L.A., Koper, E.J.R., & Giesbertz, W.J., (2004). Work processes for the development of integrated e-learning courses. In W. Jochems, J. van Merriënboer, & E.J.R. Koper, Integrated eLearning (pp. 126-138). London: RoutledgeFalmer.

Sloep, P.B. (2004a) *Leerobjecten voor gedistribueerde leeromgevingen* [Learning Objects for Distributed Learning Environments] Inaugural Address. (Eindhoven, Fontys Hogescholen).

Sloep, P.B. (2004b) Reuse, portability and interoperability of learning content: Or, why educational modelling languages? In: R. McGreal (ed.) Online education using learning objects. London, New York: Routledge Falmer. pp. 128-138.

Sloep, P.B., Hummel, H., and Manderveld, J. (2005) *Designing instruction and learning with IMS Learning Design*. In: Koper, R. & C. Tattersall (Eds) *Learning Design; A Handbook on Modelling and Delivering Networked Education and Training*. Heidelberg: Springer Verlag, pp. 133-175.

Tattersall, C. H. Vogten, R. Koper (2005) An architecture for delivery of e-learning courses. In: Koper, R. & C. Tattersall (Eds) *Learning Design; A Handbook on Modelling and Delivering Networked Education and Training*. Heidelberg: Springer Verlag, pp. 55- 66.

Vogten, H., R. Koper, H. Martens, C. Tattersall (2005). An architecture for Learning Design Engines. In: Koper, R. & C. Tattersall (Eds) *Learning Design; A Handbook on Modelling and Delivering Networked Education and Training*. Heidelberg: Springer Verlag, pp. 67 – 83.

Vogten, H., Tattersall, C., Koper, R., Van Rosmalen, P., Brouns, F., Van Bruggen, J. Sloep, P., Martens, H. (submitted) Implementing a learning design engine as a collection of finite state machines.

Wilson, S. Architectures to support authoring and content management with Learning Design. In: In: Koper, R. & C. Tattersall (Eds) *Learning Design; A Handbook on Modelling and Delivering Networked Education and Training*. Heidelberg: Springer Verlag, pp. 41- 63.

ACKNOWLEDGEMENT

The authors wish to thank teachers and the numerous students who diligently filled out forms, participated in interviews, in short provided the raw data for the evaluation reports on which this paper is based.

We also wish to thank the management and staff of the Schloss Dagstuhl International Conference and Research Centre for Computer Science for providing a pleasant, stimulating and well-organised environment for the writing of this article.

	School 1	*School 2*
students	day-time, on-campus students; homogeneous group	distance, off-campus students; heterogeneous group
institute	institute for higher professional education	institute for higher, open, distance learning
developers	regular teachers with support from instructional designers and EML experts	multidisciplinary teams of content experts, EML experts, and instructional designers
tutors	different group than the developers	same group as the developers (content experts)

TABLE 1 Some significant differences between the learning philosophies and environments of schools of case 1 and 2

Result number	Question	Mean Yes	Standard Deviation No	Range Non respond.	Sample Size idem
1.6	Were sufficient numbers of PCs available?	3.6	1.2	1-5	31
1.7	Did you receive sufficient support?	24	2	4	30
1.8	Did the screen build up sufficiently fast?	3.6	1.0	1-5	31
1.9	Was the on-screen information arranged logically?	20	10	0	30
1.10	Was the on-screen information presented attractively?	14	16	O	30
1.11	How responsive was Edubox to your request to log in? (higher is better)	4.0	1.3	1-5	31
1.12	Was it clear at all times what you location was in the virtual learning environment provided by Edubox?	26	4	0	30
1.13	Were the various ways in which to search with key words clear to you?	12	18	0	30
1.14	Using Edubox is self-explanatory.	2.9	1.0	1-5	31
1.15	Using Edubox is fun.	2.8	0.8	1-5	31
1.16	Navigation in Edubox is easy.	12	17	0	29

TABLE 2 Selected questions from respondents. Two types of questions (a score on a five point scale, where higher is better, and yes-no questions) are reported in the same table (adapted from Appendix 1 in Janssen en Van der Klink, 1999)